| Notification Name | Interface Requirements Mapping |
|---|---|
| lnpNPAC-SMS-Operational-Information | This notification is used to support the reporting of NPAC SMS scheduled down time. This notification can be issued from the lnpNPAC-SMS object on the NPAC SMS to a SOA via the SOA to NPAC SMS interface or from the NPAC SMS to the Local SMS via the NPAC SMS to Local SMS interface. |
| subscriptionAudit-DiscrepancyRpt | This notification is used to support the reporting of audit discrepancies found during audit processing. This notification can be issued from an audit object on the NPAC SMS to a SOA via the SOA to NPAC SMS interface. |
| subscriptionAudit-Results | This notification is used to support the reporting of audit processing results. This notification can be issued from an audit object on the NPAC SMS to a SOA via the SOA to NPAC SMS interface. |
| subscriptionVersionCancellationAcknowledgeRequest | This notification is issued to new and old service providers to request that a cancellation acknowledgment be sent for a subscriber version in a cancel-pending state. This notification is issued via the SOA to NPAC SMS interface from the NPAC subscription version object if the service provider fails to acknowledge the cancellation after a tunable amount of time specified in the NPAC SMS service data table. |
| subscriptionVersionDonorSP-CustomerDisconnectDate | This notification informs the donor service provider SOA that a subscription version is being disconnected. This notification is issued from a subscription version object on the NPAC SMS to a SOA via the SOA to NPAC SMS interface. |
| subscriptionVersionLocalSMS-ActionResults | This notification contains the results of a subscriptionVersionLocalSMS-Create action once all the create requests have been attempted. It is issued from the Local SMS to the NPAC SMS via the NPAC SMS to Local SMS interface. |
| subscriptionVersionNew-NPA-NXX | This notification informs the Local SMS of a pending subscription version involving a new NPA-NXX. |
| subscriptionVersionNewSP-CreateRequest | This notification is issued to the new service provider to request that a create request be sent for the subscriber version created by the old service provider to provide authorization and/or porting information. This notification is issued via the SOA to NPAC SMS interface from the NPAC subscription version object if the new service provider failed to authorize porting of a number after a tunable amount of time specified in the NPAC SMS service data table. |

| Notification Name | Interface Requirements Mapping |
|---|---|
| subscriptionVersionOldSP-ConcurrenceRequest | This notification is issued to the old service provider to request that a create request be sent for the subscriber version created by the new service provider to provide concurrence for porting. This notification is issued via the SOA to NPAC SMS interface from the NPAC subscription version object if the old service provider failed to authorize porting of a number after a tunable amount of time specified in the NPAC SMS service data table. |
| subscriptionVersionStatusAttributeValueChange | This notification is issued when the subscription version status is modified. This notification is issued from both the NPAC SMS to Local SMS interface and the SOA to NPAC SMS interface from the subscriptionVersionNPAC object. |

## 4.2. Scoping and Filtering Support

The following section defines the scoping and filtering support for both the SOA to NPAC SMS interface and LSMS to NPAC SMS interface.

### 4.2.1. Scoping

The LSMS to NPAC SMS interface does not support scoping of CMIP operations of any type by the LSMS for the following objects:

- root
- lnpLocal-SMS
- lnpNetwork
- any object with an "empty" filter

In addition, the SOA to NPAC SMS does not support scoping of CMIP operations of any type for the following objects:

- lnpNPAC-SMS
- lnpServiceProvs

Scoped operations for subscriptionVersions to the LSMS must be supported on the baseObject (level 0) or from the lnpSubscriptions object with a non-empty filter.

The limit in scoping and functionality prevents both the NPAC and the LSMS systems from having to implement functionality or respond to large requests that are not necessary to support LNP over the mechanized interfaces.

### 4.2.2. Filtering

Filtering on the NPAC SMS is supported as defined in the GDMO. The NPAC SMS requires the Local SMS to support at a minimum the filter criterias specified below.

***Limitations:***

- OR and NOT filter support is~~are~~ not required for the Local SMS~~ or the NPAC SMS~~.
- NOT filter support is not required for the NPAC SMS.
- Filtering requests with a scope will not be issued to the Local SMS by the NPAC SMS for any object other than the subscriptionVersion object.
- Filter requests must follow the rules of the NPAC SMS. For example, a query for data that a service provider is not authorized to view will be failed with a reason of access denied.
- The NPAC will roll-back any transaction that fails. Thus, if a SOA sends a M-SET request for a range of subscription objects and one object fails, the entire operation will be failed. The NPAC SMS will return a linked reply of error results. All errors will be accessDenied except the one that caused the failure; it will be set to an appropriate error.

The following table shows the CMISE primitive filtering support required of the Local SMS by the NPAC SMS for the subscriptionVersion object.

*Exhibit 12 - CMISE Primitive Filtering Support*

| CMISE Primitives | Filter Supported | Notes |
|---|---|---|
| M-ACTION | N | No actions are defined for the subscriptionVersion object. |
| M-GET | Y | TN Range with greatOrEqual, lessOrEqual, equality must be supported for auditing. |
| M-SET | Y | TN Range with greatOrEqual, lessOrEqual, equality must be supported for Mass Update or TN range modify requests. |
| M-DELETE | Y | TN Range with greatOrEqual, lessOrEqual, equality will be supported for range disconnect or port to original requests. |

### 4.3. ~~lnpLocal-SMS-Name and lnpNPAC-SMS-Name Values~~lnpLocal-SMS-Name and lnpNPAC-SMS-Name Values

~~The following table (Exhibit 13) shows the values to be used~~The following table (Exhibit 13) shows the values to be used for all currently identified NPAC regions for lnpNPAC-SMS-Name in the lnpNPAC-SMS object. The lnpLocal-SMS-Name for the lnpLocal-SMS object will be the service provider ID followed by a dash and the lnpNPA-SMS Name (*e.g.*, 9999-Midwest Regional NPAC SMS).

*Exhibit 13 - ~~Defined lnpLocal-SMS-Name and lnpNPAC-SMS-Name Values~~Defined lnpLocal-SMS-Name and lnpNPAC-SMS-Name Values*

| NPAC SMS Region | lnpNPAC-SMS-Name |
|---|---|
| Mid-Atlantic | Mid-Atlantic Regional NPAC SMS |
| Midwest | Midwest Regional NPAC SMS |
| Northeast | Northeast Regional NPAC SMS |
| Southeast | Southeast Regional NPAC SMS |
| Southwest | Southwest Regional NPAC SMS |
| Western | West Regional NPAC SMS |
| West Coast | West Coast Regional NPAC SMS |

# 5. Secure Association Establishment

5

## 5.1. Overview

This chapter describes the security, the association management and recovery procedures for the service provider SOAs and Local SMSs to follow, and how error information will be passed between interfaces.

The first section describes the security and authentication procedures used in the NPAC SMS interface. The second section describes the NPAC SMS's behavior and error handling and suggests how a service provider SOA or Local SMS should proceed when establishing an association.

## 5.2. Security

This section describes the security processes and procedures necessary for service provider SOA systems and Local SMSs to establish a secure association and maintain secure communication with the NPAC SMS. Security threats to the NPAC SMS include:

- Spoofing - An intruder may masquerade as either the SOA, Local SMS, or NPAC SMS to falsely report information.
- Message Tampering - An intruder may modify, delete, or create messages passed.
- Denial or Disruption of Service - An intruder may cause denial or disruption of service by generating or modifying messages.
- Diversion of Resources - An intruder may generate or modify messages that cause resources to be diverted to unnecessary tasks.
- Slamming - An intruder may generate or modify messages that cause customer's service to be moved between service providers.

Security threats are prevented in the NPAC SMS by use of the following methods:

- Strong two way authentication at association.
- Insuring data integrity by detection of replay, deletion, or modification to a message.
- Insuring non-repudiation of data by guaranteeing integrity and supporting data origination authentication for each incoming message.
- Implementation of access control and application level security that allows only authorized parties to cause changes to the NPAC SMS database.

### 5.2.1. Authentication and Access Control Information

The following access control information definition will be used in the AccessControl field of the association and CMIP PDUs to insure a secure communication for both the SOA to NPAC SMS interface and the NPAC SMS to Local SMS interface:

```
LnpAccessControl ::= SEQUENCE {
    systemId            [0]  SystemID,
    systemType          [1]  SystemType,
    userId              [2]  GraphicString60 OPTIONAL,
    listId              [3]  INTEGER,
    keyId               [4]  INTEGER,
    cmipDepartureTime   [5]  GeneralizedTime,
    sequenceNumber      [6]  INTEGER (0...4294967295),
    function            [7]  AssociationFunction,
    recoveryMode        [8]  BOOLEAN
    signature           [9]  BIT STRING


}

ServiceProvId ::= GraphicString4

SystemID ::= CHOICE {
    serviceProvID [0] ServiceProvId,
    npac-sms [1] GraphicString60
}

SystemType ::= ENUMERATED {
    soa(0),
    local-sms(1),
    soa-and-local-sms(2),
    npac-sms(3) --value is only valid for AccessControl
          definition
}

AssociationFunction ::= SEQUENCE {
    soaUnits [0] SoaUnits,
    lsmsUnits [1] LSMSUnits
}
SoaUnits ::= SEQUENCE {
    soaMgmt [0] NULL OPTIONAL,
    networkDataMgmt [1] NULL OPTIONAL
}
```

*Exhibit 14 Access Control*

5.2.1.1. System Id

The system Id is the unique Id for the system using an interoperable interface and must be specified in the systemId field. For a service provider using the SOA and/or Local SMS interfaces, this is the Service Provider ID. For the NPAC SMS, it is the unique identifier for the regional SMS.

5.2.1.2. System Type

The system type that indicates the type of system using the interoperable interface must be specified in the systemType field. The valid types are SOA and/or Local SMS and NPAC SMS.

5.2.1.3. User Id

The user Id of the user of the interface can optionally be specified in the userId field for the SOA interface. This is the 60 character graphics string user identifier for a user on a SOA system. It is not validated on the NPAC SMS, however, it is used for logging purposes.

5.2.1.4. List Id

The list Id must be specified as an integer in the listId field to identify a key list. This key list is one of the key lists exchanged outside of the interface process that is known to both the NPAC SMS and the Local SMS or SOA system it is communicating with.

5.2.1.5. Key Id

The key Id of a key in the key list must be specified as an integer in the keyId field. This uniquely identifies the key in the key list used to create the digital signature. The size of the modulus for the key is 600 bits as specified by the ICC.

Keys will be treated independently at the presentation layer for an association. By using presentation layer support of a key, SOAs and LSMS systems could have unique keys. In addition, if an LSMS is made up of two processes, one supporting network subscription data and the other supporting query; they could have unique keys.

5.2.1.6. CMIP Departure Time

The CMIP departure time must be specified in GeneralizedTime in the cmipDepartureTime field as the time the PDU departed the sending system. In order to insure data integrity and no-repudiation the NPAC SMS system must be synchronized to within two minutes of the Local SMS and SOA systems that it communicates.

5.2.1.7. Sequence Number

The sequence number is a 32 bit integer that must be specified in the sequenceNumber field. It should be specified as zero at association time and incremented by one for every message sent over the association. Once the sequence number reaches 4294967295 the counter will be reset to one for the association. Please note that each sender independently keeps its own counter

for the sequence number of messages sent and received. For example, after association is established, a Local SMS could send three messages to the NPAC SMS with sequence numbers 1, 2, and 3 respectively. The NPAC SMS when sending it's first message to the Local SMS would use sequence number 1 not sequence number 4.

5.2.1.8. Association Functions

The Association Function(s) must be specified on the initial association request (AARQ PDU). The following table lists the possible Association Functions that can be specified for each of the Association Request Initiators:

*Exhibit 15 Association Functions*

| Association Function \ Association Request Initiator | SOA | Local SMS |
|---|---|---|
| **SOA Management (Audit and Subscription Version)**<br>**Classes:**<br>**lnpSubscriptions**<br>**subscriptionAudit**<br>**subscriptionVersion**<br>**subscriptionVersionNPAC** | X | |
| **Service Provider and Network Data Management**<br>**Classes:**<br>**lnpNetwork**<br>**lnpNPAC-SMS**<br>**lnpServiceProvs**<br>**lsmsFilterNPA-NXX**<br>**serviceProv**<br>**serviceProvLRN**<br>**serviceProvNetwork**<br>**serviceProv-NPA-NXX** | X | X |
| **Network and Subscription Data Download**<br>**Classes:**<br>**lnpNetwork**<br>**lnpSubscriptions** | | X |
| **Query**<br>**Classes:**<br>**All** | | X |

Note that the multiple Association Functions can be specified for an association. For example, a Local SMS can establish an association for both the process audit and network and subscription data download association functions.

5.2.1.9. Recovery Mode

The recovery mode flag is set to TRUE when a Local SMS is establishing a connection after a downtime. This flag indicates to the NPAC SMS to hold all current transactions until the Local SMS sends the Recovery Complete action. Once an association is established in recovery mode, the Local SMS should request subscription and network downloads. After these steps are complete, the Local SMS should submit the Recovery Complete action. The NPAC SMS will respond with all updates since association establishment and then normal processing will resume. See *Chapter 0, Section 0, **Sequencing of Events on Initialization/Resynchronization of Local SMS***.

The recovery mode flag applies only to the Network and Subscription Data Download Association Function.

5.2.1.10. Signature

The signature field contains the MD5 hashed and encrypted systemId, the system type, the userId, the cmipDepartureTime, and sequenceNumber without separators between those fields or other additional characters. Before hashing and encryptions, character fields are ASCII format and integer fields are 32 bit big endian. Encryption is done using RSA encryption using the key from the key list specified. Validation of this field insures data integrity and non-repudiation of data.

5.2.2. Association Establishment

Strong two way authentication at association is done for both the SOA to NPAC SMS interface and the NPAC SMS to Local SMS interface. This secure association establishment is done at the application level using the access control field described above. The access control information used during association set-up is sent in the association control messages. Association establishment can be done by the SOA to NPAC SMS or Local SMS to NPAC SMS. The NPAC SMS cannot initiate an association. The initiator of the association specifies its information in the AARQ PDU message and the responder in the AARE PDU.

The following is an example of the information exchanged in the AARQ and AARE PDUs and the processing involved. Assume for the example:

- A Local SMS is making an association with the NPAC SMS.
- The Local SMS systemId is "9999."
- The NPAC SMS systemId is "NPAC SMS User Id."
- The listId for the key list is 1.
- The keyId is 32.
- The key in listId 1 with a keyId of 32 is "ABC123."
- The sequence number is 0 (as required).

The Local SMS initiates the association request by creating and sending an AARQ PDU to the NPAC SMS. This AARQ PDU contains the following access control information in the syntax described above:

- The systemId of "9999."
- The listId of 1.
- The keyId of 32.
- The current Local SMS GMT time in the cmipDepartureTime.
- A sequence number of 0.
- The signature contains MD5 hashed and encrypted systemId, systemType, userId, cmipDepartureTime, and the sequenceNumber using the encryption key "ABC123" as found in key list 1 with key id 32.
- And all BOOLEAN items are set to FALSE in the functional groups field, except for the LSMSUnit of Query item which is set to TRUE.

Once the AARQ PDU is sent, the sender (in this case the Local SMS), starts a tunable timer (with a default value of 2 minutes). If the timer expires before the AARE PDU is received then the Local SMS will terminate the association attempt.

When the NPAC SMS receives the association request it validates the data received. The data is validated as follows:

- Insure the systemId is present and valid for the association.
- Insure the sequence number is 0.
- Insure the cmipDepartureTime is within 5 minutes of the current NPAC SMS GMT time.
- Find the key specified and decrypt the signature insuring that the systemId, systemType, userId, cmipDepartureTime, and sequenceNumber are the same as those specified in the PDU.
- The functional groups requested are valid for the system type that requested the association. In this example, the system type must be "local-sms(1)" or "soa-and-local-sms(2)."

If validation of the AARQ PDU fails then an A-ABORT will be issued by the NPAC SMS with an error of access denied. If the validation of the AARQ PDU is successful then an AARE PDU would be sent back to the Local SMS. This AARE PDU contains the following access control information in the syntax described above:

- The systemId of "NPAC SMS User Id."
- The listId of 1.
- The keyId of 32.
- The current NPAC SMS GMT time in the cmipDepartureTime.
- A sequence number of 0.
- And the signature contains MD5 hashed and encrypted systemId, systemType, userId, cmipDepartureTime, and the sequenceNumber using the encryption key "ABC123" as found in key list 1 with key id 32.

The NPAC SMS may choose to optionally specify a new listId and keyId if for any reason it wants to make a key change. When the Local SMS receives the association response it validates the data received. The data is validated as follows:

- Insure the systemId is present and valid for the association. (Note: the userId field is not required for Local SMS and NPAC SMS associations).
- Insure the sequence number is 0.
- Insure the cmipDepartureTime is within 5 minutes of the current Local SMS GMT time.
- Find the key specified and decrypt the signature insuring that the systemId, systemType, userId, cmipDepartureTime, and sequenceNumber are the same as those specified in the PDU.

If validation of the AARE PDU fails then an A-ABORT will be issued by the Local SMS. If validation is successful then an secure association has been established.

### 5.2.3. Data Origination Authentication

For M-GET, M-SET, M-CREATE, M-DELETE, and M-ACTION, the access control field described above is used for data origination authentication. Please note that any of the messages sent between manager and agent must be sent in confirmed mode. The following is an example of the information exchanged in the CMIP PDUs and the processing involved. Assume for the example:

- A Local SMS is making an association with the NPAC SMS.
- The Local SMS systemId is "9999."
- The NPAC SMS systemId is "NPAC SMS User Id."
- The listId for the key list is 1.
- The keyId is 32.
- The key in listId 1 with a keyId of 32 is "ABC123."
- The sequence number is 1.

The Local SMS sends an M-GET to the NPAC SMS. The M-GET PDU contains the following access control information in the syntax described above:

- The systemId of "9999."
- The listId of 1.
- The keyId of 32.
- The current Local SMS GMT time in the cmipDepartureTime.
- A sequence number of 1.
- And the signature contains MD5 hashed and encrypted systemId, systemType, userId, cmipDepartureTime, and the sequenceNumber using the encryption key "ABC123" as found in key list 1 with key Id 32.

Once the M-GET is sent, the sender (in this case the Local SMS), starts a tunable timer (with a default value of 2 minutes). If the timer expires before the M-GET CMISE service r~~R~~esponse is received then the Local SMS will regenerate the sequenceNumber, cmipDepartureTime and signature and resend the request. The Local SMS should resend 3 times and abort the association if no response is received. If a response is received after the timeout period, it should be discarded. If an error message is received on a retry request, it should be evaluated to see if the request was processed or the error was received for other reasons. For example, an error of "duplicateObjectInstance" for an M-CREATE request most likely indicates a successful create.

When the NPAC SMS receives the M-GET request it validates the data received. The data is validated as follows:

- Insure the systemId is present and valid for the association. (Note: the userId field is not required for Local SMS and NPAC SMS associations).
- Insure the sequence number is the next sequence number expected. (In this case 1).
- Insure the cmipDepartureTime is within 5 minutes of the current NPAC SMS time.
- Find the key specified and decrypt the signature, insuring that the systemId, systemType, userId, cmipDepartureTime, and sequenceNumber are the same as those specified in the PDU.

If validation of the M-GET PDU fails then an A-ABORT will be issued by the NPAC SMS without any additional information to prevent tampering and unauthorized use of network resources by intruders. If the validation of the M-GET PDU is successful then the NPAC SMS would get the data requested and send an M-GET Response would be sent back to the Local SMS.

Since CMIP notifications (M-EVENT-REPORT) do not have access control fields, all notifications defined contain the access control information in the notification definition. ObjectCreation, ObjectDeletion, and AttributeValueChange should use the "information" attribute (*i.e.*, sub-index 6.1.7.3, 7.1.6.3, and 8.1.6.3 in section 9.21.5, *subscriptionVersionNPACNotifications*, Exhibit 83), which is an ANY DEFINED BY to contain the access control field. The values and authentication for the notification access control fields are the same as above.

### 5.2.4. Audit Trail

Audit trails will be maintained in logs on the NPAC SMS for the following association information:

- Association set-up messages.
- Association termination messages.
- Invalid messages:
  - Invalid digital signature.
  - Sequence number out of order.
  - Generalized time out of range.
  - Invalid origination address.
- All incoming messages regardless of whether or not they cause changes to data stored in the NPAC SMS.

This information will be made available for report generation on the NPAC SMS system. It will not be made available through the NPAC SMS Interoperable Interface.

## 5.3. Association Management and Recovery

### 5.3.1. Establishing Associations

#### 5.3.1.1. NpacAssociationUserInfo

The following structure will be used to report the status of a login attempt or the current state of the NPAC SMS:

```
NpacAssociationUserInfo ::= SEQUENCE {
   error-code [0] IMPLICIT ErrorCode,
   error-text [1] IMPLICIT GraphicString(SIZE(1..80))
}

ErrorCode ::= ENUMERATED
{
   success (0),
   access-denied (1)
   retry-same-host (2)
   try-other-host (3)
}
```

Bind Requests and Responses

For AARQ (M-Bind requests) the NPAC SMS will be ignoring the CMIPUserInfo userInfo field. The SMASEUserInfo will be ignored by the NPAC SMS.

In order to validate a successful login, the AARE (M-Bind response) from the NPAC SMS will contain the NpacAssociationUserInfo as the "userInfo" field of the CMIPUserInfo that is contained on the AARE. The ErrorCode will be set to "success".

The following structure will be used for CMIPUserInfo:

```
CMIPUserInfo ::= 2:9:1:1:4
--{joint-iso-ccitt(2) ms(9) cmip(1) cmip-pci(1)
abstractSyntax(4)}

CMIPUserInfo ::= SEQUENCE {
   protocolVersion [0] IMPLICIT ProtocolVersion
   DEFAULT {version1-cmip-assoc},
   functionalUnits [1] IMPLICIT FunctionalUnits DEFAULT {},
   accessControl   [2] EXTERNAL OPTIONAL
```

```
    userInfo        [3] EXTERNAL OPTIONAL
}
```

5.3.1.2. Unbind Requests and Responses

The NPAC SMS will never be issuing the RLRQ (M-Unbind request), but will respond to them from the SOA or Local SMS.

5.3.1.3. Aborts

For unsuccessful logon attempts or situations where the NPAC SMS application must abort all associations, the ABRT CMIPAbortInfo structure's "userInfo" will contain the NpacAssociationUserInfo structure. The ErrorCode will be set to one of the enumeration values.

The following structure will be used for CMIPAbortInfo:

```
CMIPAbortInfo ::= 2:9:1:1:4
--{joint-iso-ccitt(2) ms(9) cmip(1) cmip-pci(1)
abstractSyntax(4)}

CMIPAbortInfo ::= SEQUENCE {
   abortSource [0] IMPLICIT CMIPAbortSource,
   userInfo    [1] EXTERNAL OPTIONAL
}
```

5.3.1.4. NPAC SMS Behavior

Under normal conditions, the primary NPAC SMS will be responding by accepting association requests while the secondary NPAC SMS will be responding by denying association requests with an ABRT and error code of TRY_OTHER_HOST.

When the primary NPAC SMS needs to go down for a short period of time (secondary will not take over), the primary NPAC SMS will either not be responding (if down) or be denying association requests with an error code of RETRY _SAME_HOST (if partially up). The secondary NPAC SMS will be responding by denying association requests with an ABRT and error code of TRY_OTHER_HOST.

When the primary NPAC SMS goes down (scheduled or unscheduled) and the secondary NPAC SMS is re-synchronizing to become active, the primary NPAC SMS will be denying association requests with an ABRT and error code of TRY_OTHER_HOST. The secondary NPAC SMS will be responding by denying association requests with an ABRT and error code of RETRY_SAME_HOST. Once the secondary NPAC SMS is done re-synchronizing, it will then start accepting association requests.

5.3.1.5. Service Provider SOA and Local SMS Procedures

The following is an algorithm that can be used by a service provider SOA or Local SMS when trying to establish an association with the NPAC SMS:

```
        #
        #   wait X seconds
        #   execute this algorithm again substituting
        #   "secondary" for "primary"
    }
```

5.3.2. Releasing or Aborting Associations

Any of the systems, NPAC SMS, service provider SOA or Local SMS can abort an association at any time. Only the SOA and Local SMS can perform an RLRQ request. Once a scheduled outage has arrived, the NPAC SMS will abort associations (error code of "Try Other Host" or "Retry Same Host" depending on the type of outage).

5.3.3. Error Handling

5.3.3.1. NPAC SMS Error Handling

The NPAC SMS will issue errors to the Local SMS and SOA interfaces based upon the definitions and mappings in Appendix A. The NPAC SMS expects the SOA and Local SMS to support the same error definitions when both issuing and receiving error responses for the operations each interface supports.

The NPAC SMS will attempt to interpret an error returned from a SOA or Local SMS. The NPAC SMS will either retry a tunable number of times or the error will be logged. If the request is not resent and the error response was returned from a Local SMS and related to a subscription version broadcast (M-CREATE or Create Action, M-DELETE, M-SET), a broadcast failure will be noted for the service provider on the subscription version. If a service provider does not have an active Local SMS association at the time of a broadcast, the broadcast will be automatically failed for the service provider.

The Local SMS and SOA are expected to re-synchronize themselves with the NPAC SMS when their association is reestablished. Thus it is the responsibility of the Local SMS and SOA to request the necessary data to rectify the failed transmission of M-EVENT-REPORTs, network data updates and non-broadcast oriented subscription version updates. Subscription version broadcast updates to the Local SMS can be re-transmitted.

If the NPAC SMS sends a request to a Local SMS or SOA and receives no response from the CMISE service within the tunable~~timeout~~ period, the NPAC SMS will resend the message according to the tunable retry periods for the specific message type. If a response is received after the timeout period, it will be discarded. If the NPAC SMS receives no response, the NPAC SMS will assume the association is down and abort the connection. The Local SMS and SOA systems should assume the same behavior with the NPAC SMS.

5.3.3.2. Processing Failure Error

In addition to the standard CMIP error reporting mechanisms, the following attribute will be passed in the SpecificErrorInfo structure on CMIP errors that return a PROCESSING FAILURE error. This structure will be used to detail errors not covered by the standard CMIP error codes.

```
GDMO Definition

lnpSpecificInfo ATTRIBUTE
```

```
    WITH ATTRIBUTE SYNTAX LNP-ASN1.LnpSpecificInfo;
    MATCHES FOR EQUALITY;
    BEHAVIOUR lnpSpecificInfoBehavior;
    REGISTERED AS {lnp-attribute 8};

lnpSpecificInfoBehavior BEHAVIOUR
    DEFINED AS !
       This attribute is used to return more detailed
       error text information upon a CMIP Processing
       Failure error.
!;
```

**ASN.1 Definition**

```
LnpSpecificInfo ::= GraphicString(SIZE(1..256))
```

5.3.4. Resynchronization

The SOA and Local SMS associations are viewed to be permanent connections by the NPAC SMS. Thus when the association is broken for any reason, the system connecting to the NPAC SMS must assume responsibility to resynchronize themselves with the NPAC SMS. One association should be established for recovery and no other associations should be established in normal mode until recovery is complete.

5.3.4.1. Local SMS Resynchronization

To resynchronize itself, the Local SMS starts by setting the recoveryMode flag of the access control parameter. This flag signals the NPAC SMS to hold all data updates to this Local SMS. The Local SMS should then request the downloads it needs. Once this is complete, the Local SMS should issue the lnpRecoveryComplete action to turn off the recoveryMode flag and receive back any other updates that have occurred since the association was established.

5.3.4.2. SOA Resynchronization

The SOA interface resynchronizes itself by issuing the necessary queries that inform it of updates made to objects it is concerned with since it last had an association with the NPAC SMS. For subscription objects, a query should be launched based upon the new or old service provider equal to the SOA service provider and the subscriptionModifiedTimeStamp to be greater than the time when the association was lost.

Audit results may only be viewed from the NPAC SMS GUI and are not available on the mechanized interface.

# 6. Message Flow Diagrams

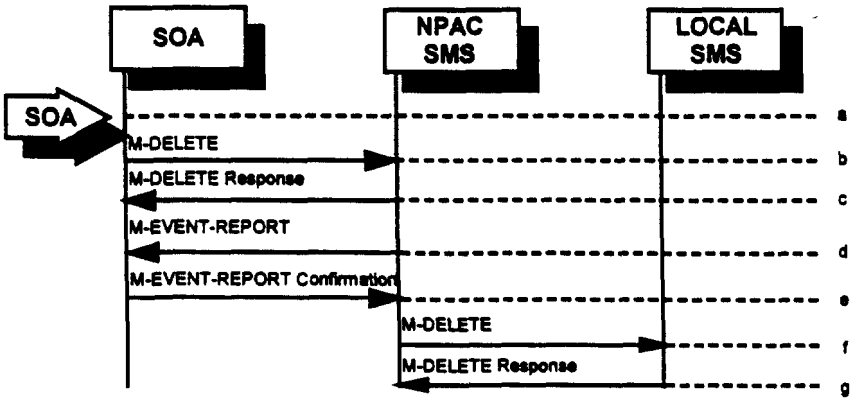
6

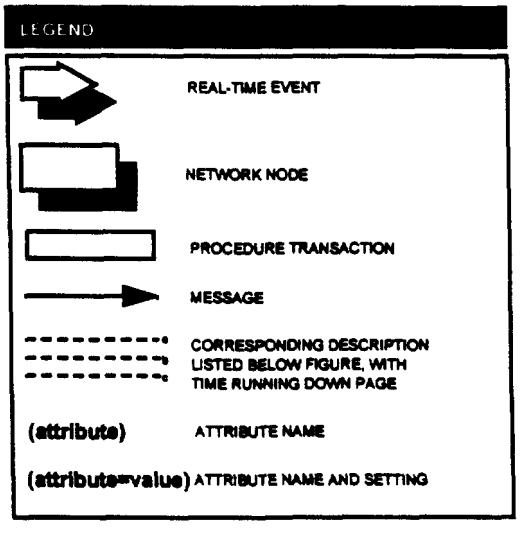
## 6.1. Overview

This chapter defines the message flow scenarios for the SOA to NPAC and the NPAC SMS to Local SMS interfaces. Each of these definitions consists of a message flow diagram and a textual description of the diagram.

NOTE: The order of messages in the message flows must be followed by the NPAC SMS, SOA, and LSMS systems with the exception of the return of the M-EVENT-REPORT confirmations.

The following is an example message flow diagram and legend for elements shown in the diagram.

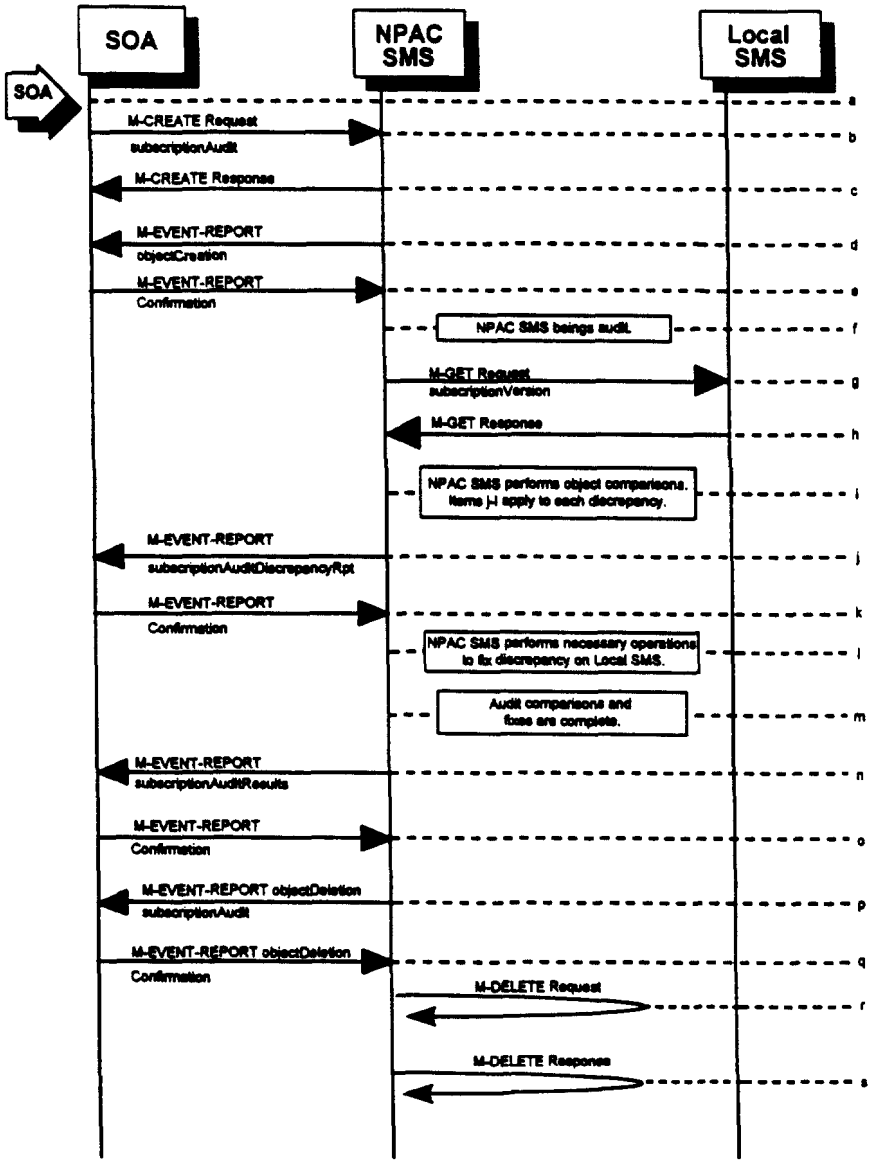


## 6.2. Audit Scenarios

### 6.2.1. SOA Initiated Audit

In this scenario, the SOA initiates an audit to the NPAC SMS due to suspected subscription version discrepancies.




a. Action is taken by SOA personnel to start an audit due to suspected network discrepancies.

b. The SOA sends a M-CREATE request to the NPAC SMS, requesting an audit. The SOA must specify the following attributes in the request:

~~**serviceProvID** - SOA service provider id~~
**subscriptionAuditName** - English audit name
**subscriptionAuditRequestingSP** - the service provider requesting the audit
**subscriptionAuditServiceProvIdRange** - which service provider or all service

providers for audit
**subscriptionAuditTN-Range** - TNs to be audited

If these attributes are not specified, then the create will fail with a missingAttributesValue error. The SOA may also specify the following attributes in the request:

**subscriptionAuditAttributeList** - subscription version attributes to be audited
**subscriptionAuditTN-ActivationRange** - time range of activation for subscription versions to be audited

The subscriptionAuditId and the subscriptionAuditStatus will be determined by the NPAC SMS. If any values are deemed invalid, an invalidArgumentValue error will be returned. **NOTE**: The subscriptionAuditTN-Range will be limited based on the maximum range size specified in the NPAC SMS. If the limit specified is exceeded, the create request will fail with an invalidAttributeValue error.

c. Once the NPAC SMS creates the audit request object, it sends an M-CREATE response back to the SOA that initiated the request.

d. NPAC SMS sends M-EVENT-REPORT to the service provider SOA for the subscriptionAudit creation.

e. The service provider SOA confirms the M-EVENT-REPORT.

f. NPAC SMS begins audit.

g. NPAC SMS issues a scoped and filtered M-GET for the subscription versions in the audit.

h. Local SMS returns M-GET query data.

i. NPAC SMS performs the necessary comparisons of each subscription version object.

j. If a discrepancy is found, NPAC SMS issues a subscriptionAuditDiscrepancyRpt M-EVENT-REPORT.

k. Service provider SOA confirms the M-EVENT-REPORT.

l. If a discrepancy is found, NPAC SMS issues the necessary operation to the Local SMS to correct the discrepancy (M-CREATE, M-DELETE, or M-SET).

m. NPAC SMS has completed the audit comparisons and corrections.

n. NPAC SMS issues the subscriptionAuditResults M-EVENT-REPORT to the service provider SOA.

o. The Service provider SOA confirms the M-EVENT-REPORT.

p. The NPAC SMS then sends an objectDeletion M-EVENT-REPORT to the SOA for the subscriptionAudit object.

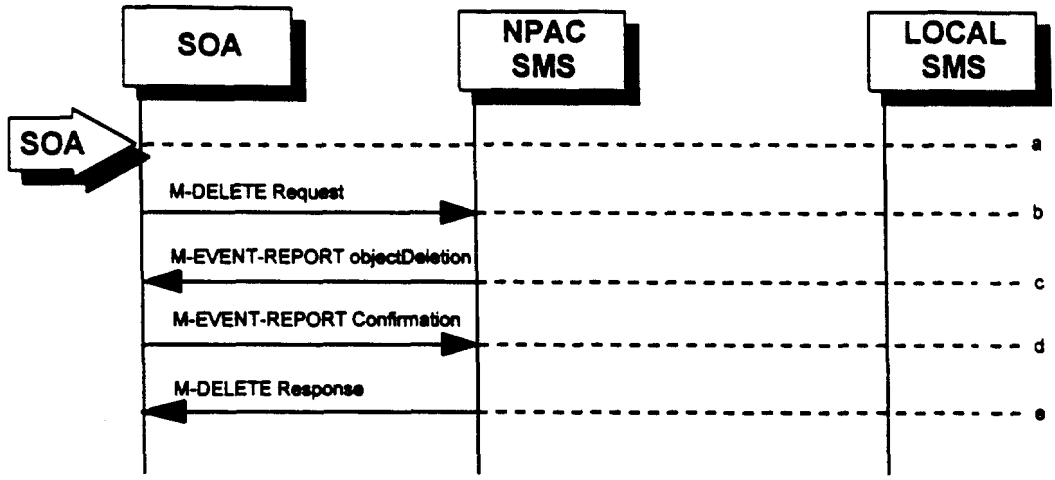
q. The service provider SOA confirms the M-EVENT-REPORT.

r. The NPAC SMS issues a local M-DELETE request for the subscriptionAudit object to/from the NPAC SMS. This will attempt to delete the subscriptionAudit object on the NPAC SMS.

s. The M-DELETE response is received on the NPAC SMS indicating whether the subscriptionAudit object was deleted successfully.

6.2.2. SOA Initiated Audit Cancellation by the SOA

The SOA cancels an audit that it initiated.




a. Action is taken by SOA personnel to cancel an audit previously initiated by the SOA.

b. The SOA sends an M-DELETE request for the subscriptionAudit object to the NPAC SMS, requesting cancellation of an audit. If the audit was not initiated by the SOA requesting cancellation, then the request will be rejected with an accessDenied error.

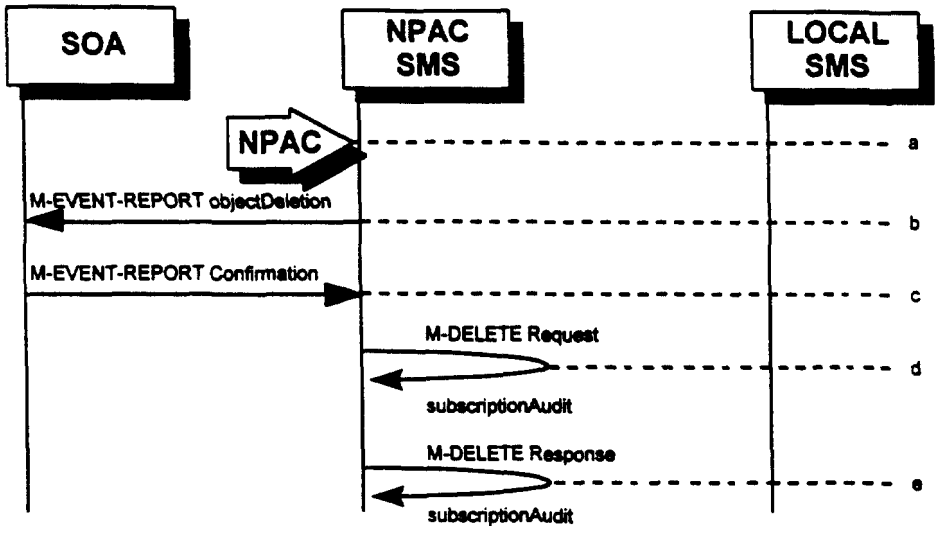
c. The NPAC SMS will respond by sending an objectDeletion M-EVENT-REPORT.

d. The SOA confirms the M-EVENT-REPORT.

e. The NPAC SMS sends an M-DELETE response to the SOA.

6.2.3. SOA Initiated Audit Cancellation by the NPAC

The NPAC cancels an audit that was initiated by an SOA.




a. Action is taken by NPAC personnel to cancel an audit previously initiated by an SOA.

b. The NPAC SMS sends an objectDeletion M-EVENT-REPORT to the SOA that initiated the audit request.

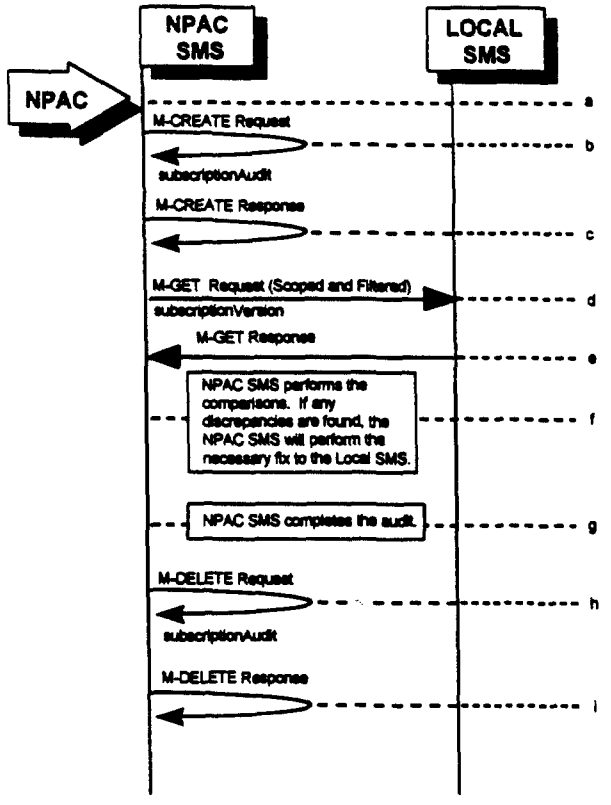
c. The SOA confirms the M-EVENT-REPORT

d. The NPAC SMS issues a local M-DELETE request to/from the NPAC SMS. This will attempt to delete the subscriptionAudit object on the NPAC SMS.

e. The M-DELETE response is received on the NPAC SMS indicating whether the subscriptionAudit object was deleted successfully.

### 6.2.4. NPAC Initiated Audit

In this scenario, the NPAC SMS initiates an audit due to suspected subscription version discrepancies.




a. Action is taken by NPAC personnel to start an audit due to suspected network discrepancies.

b. The NPAC SMS does a Local M-CREATE request to itself for the subscriptionAudit object requesting an audit.

c. The NPAC SMS responds with an M-CREATE response indicating that the subscriptionAudit object was created successfully.

d. The NPAC SMS sends an M-GET request to the Local SMSs to retrieve the subscription data to use for audit processing. The request uses the CMIP scoping and filtering options to retrieve only the subscriptionVersion objects to be audited.

e. The Local SMS responds to the M-GET request by returning the subscription data that satisfies the scope and filter data.

f. NPAC SMS performs the comparisons. If any discrepancies are found, the NPAC SMS will perform the necessary fix to the Local SMS.

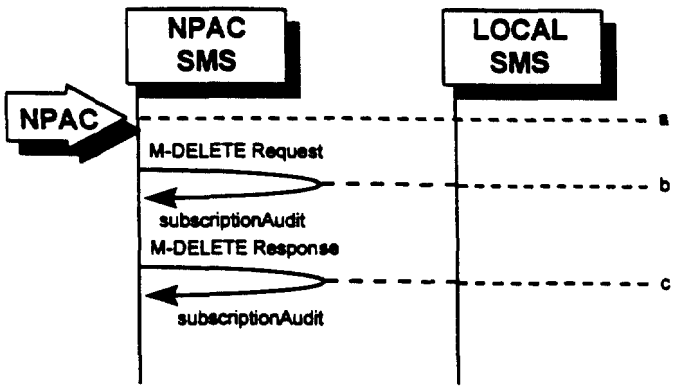
g. NPAC SMS completes the audit.

h. Issue a local M-DELETE request for the subscriptionAudit object to/from the NPAC SMS. This will attempt to delete the subscriptionAudit object on the NPAC SMS.

i. The M-DELETE response is received on the NPAC SMS indicating whether the subscriptionAudit object was deleted successfully.

6.2.5. NPAC Initiated Audit Cancellation by the NPAC

The NPAC SMS cancels an audit that it initiated.



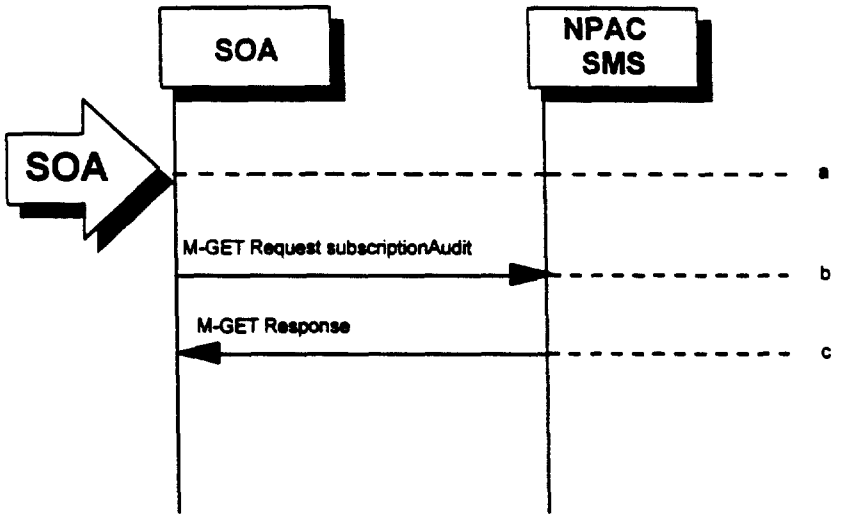

a. Action is taken by NPAC personnel to cancel an audit previously initiated by the NPAC SMS.

b. Issue a local M-DELETE request to/from the NPAC SMS. This will attempt to delete the subscriptionAudit object on the NPAC SMS.

c. The M-DELETE response is received on the NPAC SMS indicating whether the subscriptionAudit object was deleted successfully.

### 6.2.6. Audit Query on the NPAC

This scenario shows a service provider query on an existing audit that it initiated.




a. The service provider SOA takes action to query an audit that it initiated.

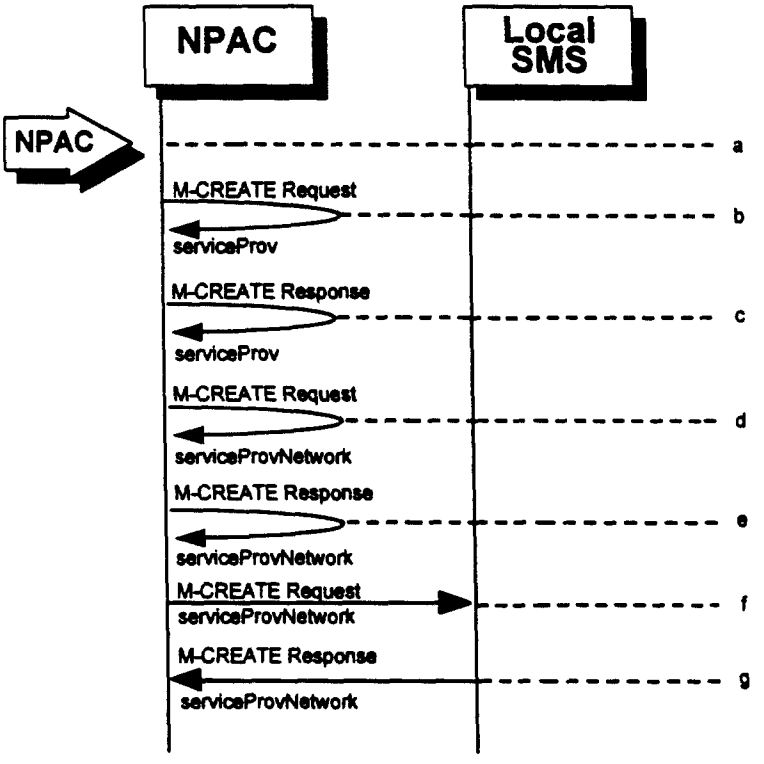
b. Service provider SOA sends an M-GET request for a subscriptionAudit on the NPAC SMS.

c. NPAC SMS responds to an M-GET with the audit data or a failure and reason for failure. An accessDenied error will be returned to the service provider if they did not originate the audit queried.

## 6.3. Service Provider Scenarios

### 6.3.1. Service Provider Creation by the NPAC

In this scenario, the NPAC SMS creates data for a new LNP service provider. The addition of NPA-NXX and LRN data for a new service provider will be shown in flows that follow.




a. Action is taken by NPAC SMS personnel to create a new service provider.

b. Issue a local M-CREATE request for the serviceProv object to/from the NPAC SMS. This will attempt to create the serviceProv object on the NPAC SMS. If the M-CREATE fails, the appropriate error will be returned.

c. The M-CREATE response is received on the NPAC SMS indicating whether the serviceProv object was created successfully. If a failure occurs, processing will stop.

d. Issue a local M-CREATE request for the serviceProvNetwork object to/from the NPAC SMS. This will attempt to create the serviceProvNetwork object on the NPAC SMS. If the M-CREATE fails, the appropriate error will be returned.

e. The M-CREATE response is received on the NPAC SMS indicating whether the serviceProvNetwork object was created successfully. If the object cannot be created, the serviceProv object is deleted and an error is returned.

f. The NPAC SMS sends an M-CREATE request for the serviceProvNetwork object to each of the Local SMSs.

g. The Local SMS(s) will respond by sending an M-CREATE response back to the NPAC SMS.